1. 描述何为多态？

允许将父对象设置成为和一个或多个它的子对象相等的技术，比如Parent:=Child； 多态性使得能够利用同一类(基类)类型的指针来引用不同类的对象,以及根据所引用对象的不同,以不同的方式执行相同的操作.

1. 实现一个函数，判断给定的括号字符串是否配对。””

语言支持的话就正则式判断不支持只能从两边开始读，左边读到{或[或(停止，然后右边读)或]或}停止，不匹配返回false，匹配继续读，直到左边指针位置=右边指针位置。

1. 实现一个函数，在给定数组中寻找第2 大的数。

def fn():

a=[1,3,5,7,9]

return sorted(a[1])

1. Point是一个类，有两个成员：int x; int y;
2. 请声明一个动态数组arry，用以存放若干 Point对象；
3. 已知一个Point currPoint，请写一个算法，将arry中所有Point中距离currPoint最近的n个Point删掉（可以用平面几何距离，这n个点不一定等距）

返回的是删除过后的新数组

1. <?php
2. function removePoint($arry, $current){
3. $d = array();
4. foreach($array as $k=>$v){
5. if(distent($current, $v)){
6. [array\_push](https://www.baidu.com/s?wd=array_push&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "_blank)($d, distent($current, $v));
7. }
8. }
9. asort($d);
10. $sk = key($d);
11. unset($array[$k]);
12. return $array;
13. }
14. function distent($a, $b){
15. if(is\_object($a) && is\_object($b)){
16. $d = sqrt(($a->x - $b->x) \* ($a->x - $b->x) + ($a->y - $b->y) \* ($a->y - $b->y));
17. return $d;
18. }
19. return false;
20. }
21. ?>

3)如果Point的数量非常巨大，在这些点中可能会存在两点距离小于d的状况，这是异常数据，请写一个算法将其全部找出。

**将point放在 x/d y/d 的hash表里。对每个变量，只要判断x-d~x+d和x-d~y+d的相邻的hash项, 两点距离可以用曼哈顿距离来模拟**

**将整个空间分割为多个“网格”，首先对每一个已知的点计算出它分布在哪一个“行、列”的网格中，建立倒排索引。当要遍历一些点的时候，你可以先找到所要查询的中心点周边有多少网格（例如9宫格），然后按照索引来遍历这些网格中的点，而不用去遍历整个的点集。  
  
这样做，最主要地好处不是在于其精确索引、而是在于其“粗放”的特性。因为按照“x>a 并且x<=a+m并且y>b 并且y<=b+m”的查询方式，当频繁查询时，这里的(a,b)是精确的坐标，不适合作为 cache 的 key。而如果把(a,b)变为网格的“行,列”编号则可以大大减少缓存单元的数量，提高缓存命中率。  
  
因此在考虑每天上亿次频繁查询时，可以采用更为粗放的网格排序，更好地使用缓存技术。**